REMARKS

Applicant appreciates the time taken by the Examiner to review Applicant's present application. This application has been carefully reviewed in light of the Official Action mailed on March 31, 2008, the Examiner Interview on May 15, 2008, and the Interview Summary mailed on May 28, 2008. This Reply encompasses a bona fide attempt to fully respond to the Examiner's rejections, including amendments as well as reasons why Applicant believes that embodiments of the invention as claimed are patentable over the applied prior art. Applicant respectfully requests reconsideration and favorable action in this case.

Interview Summary

Pursuant to Applicant Initiated Interview Request submitted on May 2, 2008, a telephonic interview was conducted on May 15, 2008 between Examiner Mahesh Dwivedi and Attorney Katharina Schuster. Differences between embodiments as claimed and the cited prior art references were discussed. In particular, possible amendments to Claims 37 and 55 with respect to filesystem architecture were discussed. Examiner Dwivedi indicated that claim amendments clarifying that the filesystems are "networked" would overcome the applied prior art. Applicant appreciates the time and effort taken by Examiner Dwivedi to review Applicant's present application and discuss the pending claims and the cited prior art.

Claim Status

Claims 37 and 39-105 were pending. Claims 1-36 and 106-113 were previously withdrawn. Claim 38 was cancelled previously. Claims 37, 55, 60, 72, 89, and 94 are amended herein. Support for the amendments presented herein can be found in the Specification as originally filed, at least from paragraphs 7, 18, 64, 103, and 109. No new matter is introduced. No claim is newly added. By this amendment, Claims 37 and 39-105 remain pending.

Rejections under 35 U.S.C. § 103

Claims 37, 39-43, 52-59, 69-76, 86-93 and 103-105 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent Application Publication No. 2002/0046262 ("Heilig") in view of U.S. Patent No. 6,654,814 ("Britton"). This rejection appears to be the same as the rejection set forth in the previous Office Action mailed on November 1, 2007. Thus,

arguments submitted in the previous Reply mailed on February 1, 2008 remain pertinent and are expressly incorporated herein by reference. As indicated by the Examiner during the aforementioned Interview, claim amendments clarifying that embodiments as claimed are directed to filesystem architecture would overcome Heilig and Britton. As a good faith attempt to expedite the prosecution, independent claims 37, 55, 72, and 89 are amended herein to make explicit that which was implicit, the applicability of Heilig, Britton, and Kao to claims 37 and 39-105 notwithstanding. Specifically, as amended, independent claim 37 recites:

A system comprising:
a network;
a plurality of client computers, each client computer comprising:
a client processor;
a client network interface to connect to and interface with the network;
a client computer readable medium accessible by the client processor,
storing a client program executable by the client processor to:
generate a first filesystem request according to a first network filesystem protocol, wherein said first network filesystem protocol extends a filesystem namespace and abstractions across said network;
receive a first filesystem response;
an intermediary device comprising:
an intermediary processor;
an intermediary network interface to connect to and interface with the network;
an intermediary computer readable medium accessible by the intermediary processor and executable to:
provide a client-facing filesystem interface employing said first network filesystem protocol;
provide a server-facing filesystem interface employing a second network filesystem protocol;
receive the first filesystem request from a requesting client according to the client-facing filesystem interface;
pass the first filesystem request to a server as a proxy request according to the server-facing filesystem interface, wherein passing the first filesystem request as a proxy request comprises applying a set of rules to the first filesystem request to determine if the first filesystem request should be modified and if it is determined that the first filesystem request should be modified, modifying the first filesystem request to generate the proxy request;
receive a server response from the server according to the server facing filesystem interface;
pass the server response to the requesting client as the first filesystem response;
a plurality of servers, each server further comprising:
a server processor;
a server interface coupled to the server processor to connect to and interface with the network; and
a server computer readable medium storing a server program executable by the server processor to:
provide an origin filesystem;
receive the proxy request from the intermediary device;
execute a requested filesystem operation;

generate the server response; and
communicate the server response to the intermediary computer; and

a plurality of storage media devices, wherein each of the plurality of storage media devices is connected to and associated with one of the plurality of servers, wherein each of the plurality of storage media devices has a network filesystem that implements said second network filesystem protocol, and wherein said first network filesystem protocol is same as or different from said second network filesystem protocol.

As described in the Specification, at the time of the invention, a storage resource can be accessed by a computer over a network connection. *See* Specification, paragraphs 6-7. Various mechanisms exist that allow software or users on one computing device to access storage devices that are located on another remote computer or device. Remote storage access mechanisms generally fall into one of two classes: block-level and file-level. File-level remote storage access mechanisms extend the filesystem interface and namespace across the network, enabling clients to access and utilize the files and directories as if they were local. Such systems are therefore typically called "network file system." One Example of this type of storage access mechanism is the Network File System ("NFS") originally developed by Sun Microsystems. Within the context of the present application, the term "network filesystem" is used to refer to all such systems.

Networked filesystems enable machines to access the filesystems that reside on other machines. Architecturally, this leads to the following distinctions. In the context of a given filesystem, one machine plays the role of a filesystem "origin server" (alternatively either "fileserver" or simply "server") and another plays the role of a filesystem client. The client and server machines are connected via a data transmission network and communicate over this network using network filesystem protocols which extend the filesystem namespaces and abstractions across the network. At the time of the invention, many such protocols exist, including the Common Internet File System or CIFS, the aforementioned NFS, Novell's Netware filesharing system, Apple's Appleshare, the Andrew File System (AFS), the Coda Filesystem (Coda), and others. CFS and NFS are by far the most prevalent. All of these network filesystem protocols share approximately equivalent semantics and sets of abstractions, but differ in their details and therefore are noninteroperable. *Id.*

Embodiments as claimed in claims 37, 55, 72, and 89 can address this problem. More specifically, in embodiments as claimed in claims 37, 55, 72, and 89, the client facing filesystem interface can employ a network filesystem protocol implementation and the server facing filesystem interface can employ the same or different network filesystem protocol implementation. *See* Specification, paragraph 109. To clients, the intermediary device appears as a server and, to servers, the intermediate device appears as a client. *Id.* One advantage is that, if needed, arbitrary protocol translation and bridging between the different network filesystem protocols can be performed at the intermediary device. *Id.*

As submitted before, Heilig and Britton are not directed to solving problems pertaining to network filesystems. For example, as the Examiner pointed out on page 15 of the Office Action, Heilig teaches communications based on a bitmap protocol or X Windows protocol, citing paragraph 122 of Heilig. As those skilled in the art can appreciate, these protocols are not network filesystem protocols. By reciting, among others, that the claimed intermediary device comprises a computer readable medium executable by a processor to provide a server-facing filesystem interface employing a second network filesystem protocol and a client-facing filesystem interface employing said first network filesystem protocol, it is believed that independent claim 37 sufficiently overcomes Heilig and Britton. Independent claims 55, 72, and 89 are amended herein to recite similar limitations and are submitted to be also patentable under 35 U.S.C. § 103(a) over Heilig and Britton. Accordingly, withdrawal of this rejection is respectfully requested. .

Claims 44-51, 60-68, 77-85 and 94-102 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Heilig in view of Britton and further in view of U.S. Patent No. 5,870,734 ("Kao"). As submitted before, unlike Heilig and Britton, Kao appears to be drawn to the architectural deficiency in networked file systems (spelled as "filesystems" in the present application). Kao's solution relies on combining a three-dimensional file system with a specific architecture known as the virtual node (vnode) architecture. *See* Kao, col. 4, lines 40-62. Kao's solution does not rely on an intermediate device and appears to be similar to one of the prior attempts to address the problems of unconstrained complexity growth in the networked filesystem environment described in the present application. *See* Specification, page 8-11, *paras.* 11-16. Specifically, Kao describes that the "top directory will appear to contain a union of the contents of all of the directories in the Z-stack." *See* Kao, col. 5, line 56, through col. 6,

line 7; FIG. 1. At the time of the invention, networked filesystems do not allow arbitrary triggers and associated activities to be programmed outside of the permissions hard coded in the original implementation of the filesystems. Thus, there were no apparent reasons for one of ordinary skill in the art to modify Kao with Heilig and Britton. For the foregoing reasons and in the previous Reply submitted on February 1, 2008, Applicant respectfully requests withdrawal of this rejection.

CONCLUSION

Applicant has now made an earnest attempt to place this case in condition for allowance. Other than as explicitly set forth above, this reply does not include any acquiescence to statements, assertions, assumptions, conclusions, or any combination thereof in the Office Action. For the foregoing reasons and for other reasons clearly apparent, Applicant respectfully requests full allowance of Claims 37 and 39-105. The Examiner is invited to telephone the undersigned at the number listed below for prompt action in the event any issues remain.

The Director of the U.S. Patent and Trademark Office is hereby authorized to charge any fees or credit any overpayments to Deposit Account No. 50-3183 of Sprinkle IP Law Group.

Respectfully submitted,

Sprinkle IP Law Group
Attorneys for Applicant

Katharina Schuster

Katharina Wang Schuster
Reg. No. 50,000

Date: October 3, 2008

1301 W. 25th Street, Suite 408
Austin, TX 78705
Tel. (512) 637-9220
Fax. (512) 371-9088